Navigation Using Hybrid Genetic Programming: Initial Conditions and State Transitions

Steven J. Simske
Intelligent Enterprise Technologies Laboratory
HP Laboratories Palo Alto
HPL-2003-56
March 20th , 2003*

genetic algorithms, navigation, optimization

Real-time navigation requires the dynamic updating of node-node (state) transition costs. These state transition costs are based on, among other things, distance, traffic patterns, and intelligent clustering of nodes based on similarity in the navigator's intents at each destination, maximum distance preferred between nodes, and other pragmatic considerations. Such superimposed conditions require a hybrid genetic/rule-based system to accommodate the real-world considerations (rule-based) as well as providing efficient computation of minimized overall (summed) node-node costs (genetic algorithm-based; "traveling salesman"). This paper introduces the important elements of such a hybrid system, focusing on the use of rule-based & clustering techniques for initial conditions and node-node transition costs, while showing how genetic algorithms akin to those using gene linking can be used to efficiently compute best paths through a set of nodes with dynamic transition costs.

* Internal Accession Date Only
Approved for External Publication

Navigation Using Hybrid Genetic Programming: Initial Conditions and State Transitions

Steven J. Simske
Intelligent Enterprise Technologies Lab
HP Laboratories
March 11, 2003

ABSTRACT

Real-time navigation requires the dynamic updating of node-node (state) transition costs. These state transition costs are based on, among other things, distance, traffic patterns, and intelligent clustering of nodes based on similarity in the navigator's intents at each destination, maximum distance preferred between nodes, and other pragmatic considerations. Such superimposed conditions require a hybrid genetic/rule-based system to accommodate the real-world considerations (rule-based) as well as providing efficient computation of minimized overall (summed) node-node costs (genetic algorithm-based, "traveling salesman"). This paper introduces the important elements of such a hybrid system, focusing on the use of rule-based & clustering techniques for initial conditions and node-node transition costs, while showing how genetic algorithms akin to those using gene linking can be used to efficiently compute best paths through a set of nodes with dynamic transition costs.

Keywords: Genetic Algorithms, Navigational Systems, Traveling Salesman, Gene Linkage, Path Optimization

1 Introduction

Navigational systems [1] incorporating genetic algorithms (GA's) benefit from the utility of GA's in solving difficult optimization problems efficiently and without futile iterating around non-global minima [2]. However, when large numbers of nodes (positions that are traversed between) are involved, or when node-node transitions are difficult to represent with the traditional loci set of GA's [3], a hybrid system is required to optimize performance. Examples include navigation when particular node-node transitions either do not exist (i.e. there is no path between the two nodes) or the transitions are innately costly (e.g. crossing over within an exon, obviating its later coding of a functional protein). For navigational systems, node-node transitions can rarely be viewed in isolation, but instead practical issues around traversing particular node-node pathways must be accounted for. GA researchers have looked at gene linkage [4,5], punctuated crossover [6,7] and "messy" GA's [8] as means of allowing groups of co-adapted genes to be inherited together during the recombination/crossover phase of the GA. Multiple recombination strategies [9] can also be viewed as a means of providing for greater variety of offspring. In this paper, the concept of gene linkage via weighted node-node transition costs is introduced. Then, gene linkage is imposed directly (via initial pathway specification) and indirectly (via probabilistic crossover techniques) in the starting set of genes for two navigational problems. Finally, the ramifications of these weighted node-node costs is discussed in light of the need for "hybrid genetic programming" (combination of GA's with transitional probabilities) for navigational tasks.

2. Initial Conditions, Node-Node Transitions and Methods

Nodes herein are used to describe physical locations (points, loci). Node-node transitions are the costs associated with moving between one node and another. Typically, these are functions of distance, but herein will be simplified to simply **be** the distance. A particular node-node transition will be called a state, and any given pathway will use only a subset of all the node-node transitions, so long as the number of nodes, N, is >= 4.

Initial Conditions

Because convergence is increased when (a) the initial chromosomes are close to the optimal chromosomes, and (b) at least some of the initial chromosomes are evaluated (and selected) before the first crossover, the

following initial conditions were explored for navigation (e.g. "traveling salesman" and other test problems):

1. Naïve. Here the N nodes in a navigational pathway are chosen randomly, one after the other, until all N are exhausted. This is a reasonable starting point when N is small or when node-node distances are relatively similar for all node-node transitions.

2. Weighted Nearest Neighbor. Here, for each node, the costs to each of the other N-1 nodes is calculated, and all "#1" choices, where possible, awarded. From the remaining pool of node-node transitions, all "#2" choices, where possible, are awarded. This continues until all node-node pathways have been assigned. For example, in a 4-node problem, if the following node-node transition distances are recorded:

	Node 1	Node 2	Node 3	Node 4
Node 1	--	40	30	35
Node 2	40	--	50	55
Node 3	30	50	--	70
Node 4	35	55	70	--

Node 1-Node 3 is assigned first.
Node 1-Node 4 is assigned next.
Because Node 1 is now complete, Node 2-Node 1 cannot occur in an exhaustive complete traversal, and so the following traversal is obtained:
Node 4-Node 1-Node 3-Node 2-Node 4

3. Lowest Remaining Distance. Here, the lowest remaining distance between any "nonexhausted" nodes is always assigned. This and the preceding method are generally reasonable when the nodes are relatively evenly spaced; however, when there are a small number of outlying points, this can results in extraneous large distances to/from/between such outliers.

4. Centroid + Clockwise/Counterclockwise Traversal. For a variety of 2-D navigational tasks, it was found that selecting the starting point at random, and then following from one node to the next along a (counter)clockwise path results in an efficient initial pathway. The centroid of all points is the reference point for the rotation.

5. Clustering + Method 4. If the distances between nodes varies considerably, then the nodes themselves are candidates for clustering. Traditional clustering techniques (e.g. K-means) can be used to assign the nodes to distinct clusters, along with a method found to be useful herein for cluster definition: A cluster C composed of 2 or more nodes is valid if the minimum distance of a path between all of its nodes is less than the distance to any other nodes in the set of all nodes S.

Node-Node Transitions

Node-node transitions are weighted either to affect the initial generation of chromosomes or else to impact the location of crossover. In this paper, they are used for initial state with/without their use in crossover. Some important means of assigning node-node transition probabilities (or costs, normalized to sum to 1.0 for all possible transitions from a given node) include:

1. Naïve. Here, all probabilities are identical, and so each node-node transition cost, or probability, is defined as: $P_i = 1/(N-1)$, where N=#nodes.

2. By Relative Distance. The node-node costs are weighted by relative distance. One such method is to use a step size of ΔP, so that $P_i = P(Min)+((i-1) * \Delta P)$, where P(Min) is the minimum cost, and i=1…N.

3. By Cluster. If nodes are assigned to clusters, then the weighting within the clusters can be higher than the weighting outside of the cluster (many possible methods). That is, $p(C_i,C_j) > p(C_i,N_j)$, where C is the cluster, and N = S-C is the set of all other nodes (outside the cluster C).

4. By Direction. If a (counter)clockwise traversal is occurring, then the next nodes in the direction being traversed can be weighted more highly. This Bayesian technique is advantageous for determining dynamically the probabilities for crossover (however, due to space limitations, this is not considered further herein).

5. By Distance. An inverse weighting scheme is used here, where the weight/probability $P_i = C_i/d_{ij}$ for all nodes i to all other nodes j. Since the weighting is inversely proportional to distance, then $d_{ij}/d_{ji} = P_j/P_i$, and thus $C_j/C_i = 1$, $P_i = k/ d_{ij}$, and since $\Sigma_{i=1\ldots N}P_i = 1.0$, then $k = 1/(\Sigma_{j=1\ldots N}(1/d_{ij}))$, and so:

$$P_i = 1 / (d_i * \Sigma_{j=1\ldots N}(1/d_{ij})) \qquad \textbf{Equation 1}$$

This can be extended to any power X of distance readily as given here:

$$P_i = 1 / (d_i^X * \Sigma_{j=1\ldots N}(1/d_{ij}^X)) \qquad \textbf{Equation 2}$$

It can also be readily extended to any function f(i):

$$P_i = f(i) / \Sigma_{j=1\ldots N}(f(i))) \qquad \textbf{Equation 3}$$

In real-world applications, these node-node transition costs are based on, among other things, distance and traffic patterns between nodes, and intelligent clustering of nodes based on similarity in the navigator's intents at each destination (e.g. cities to visit may be clustered by language in Europe), maximum distance preferred between nodes, and other pragmatic considerations. These transitions can also be used to govern the initial conditions (see Results section below).

Methods

A Java-based generic genetic ("genertic") toolkit was developed, allowing for control of, among other variables, the following: (1) number of genes/chromosomes, (2) crossover rate & type, (3) mutation rate & type, (4) test for performance asymptote, (5) test for optimal fitness intransigence, (6) fitness (cost), (7) selection rate, (8) initial conditions, and (9) node-node transition costs.
The choice of each of these 9 parameters in the genertic software is dictated by the application. The real-world application chosen to represent the navigational "hybrid" genetic algorithm is the "traveling salesman" problem. Two specific test problems were considered. The first is the "Iceland" challenge, wherein 9 cities in Iceland (Anglicized to: Akranes, Akureyri, Borgarnes, Egilstaddir, Hofn, Isafjordur, Reykjavik, Selfoss and Vik, see Figure 1) were traversed in an exhaustive pathway (single traversal—all cities visited once—and return to starting point). For this type of problem, the number of possible "optimum pathways" is at least 2N, where N=#nodes, since the optimum single traversal optimum pathway can start and end at any node, and traverse either clockwise or counterclockwise. Thus, the odds of randomly selecting an optimum pathway are $2N/N! = 2/(N-1)!$ For the Iceland problem, this is 1/20160.

The second test problem used 20 cities in the United States and the reported distances between them [12]. The cities were selected to ensure that no obvious optimal pathway could be obtained (unlike the Iceland problem, in which it is in hindsight obvious that the "Centroid + Clockwise/Counterclockwise Traversal", among others, provides the optimal path, see Figure 2). For the "USA" problem, the odds of randomly selecting an optimum pathway are $2/19! = 1/(6.082255\text{x}10^{16})$. For the "Iceland" and "USA" problems, the following specifications were employed: (1) 50 genes were used (since the length is rather short, 9 or 20 loci or "nodes", they are referred to as "genes" hereafter, but could just as accurately been dubbed "chromosomes"). (2) Crossover rate was set at 90%, with crossover consisting of a roulette-wheel selection of two splicing locations (that is, two intra-loci transitions), followed by reversal of the pathways between the nodes. For example, if the original pathway was: 1234567891 for cities 1-9, and the splicing transitions were found to be 2-3 and 6-7, then our "crossed-over" result is 12**6543**7891. The Java Math.random() function was used for determining the splicing locations (e.g. 0-0.111 for nodes 1-2 splicing, 0.111-0.222 for nodes 2-3 splicing, etc.); therefore, the **true** crossover value was 80% =

(0.889*90%) due to the possibility of identical splice points for the "Iceland" problem, and 85.5% for the "USA" problem for the same reason.

Figure 1. The "Iceland" city set (cities indicated by •). Distances between cities were garnered from [10]. The map shown in Figures 1-3 was obtained from [11].

Figure 2. The "Iceland" city set with the optimal pathway (or set of 18 pathways, rather) superimposed.

(3) Mutation rate was set at 2%, and employed as a swap of two loci. For the previous example, this results in 12**6**45**3**7891. True mutation rate was 1.78% and 1.9%, respectively, for the "Iceland" and "USA" problems. (4) Rapidly-converging (asymptotic) runs of the algorithm were determined by comparing after each iteration of the GA the standard deviation (STD) of the fitness (or "cost", which was total distance) for

the first half of the iterations to the STD of the fitness over the last half of the iterations. If the ratio was greater than four, then the particular run of the algorithm was declared "converging". (5) If converging, the run was either terminated and the optimal fitness and path/s recorded or else the mutation rate was increased to 5% for one iteration and the run allowed to proceed. This "spot" increase in mutation rate was termed "jiggling." (6) Fitness cost was simply the sum of all node-node distances.

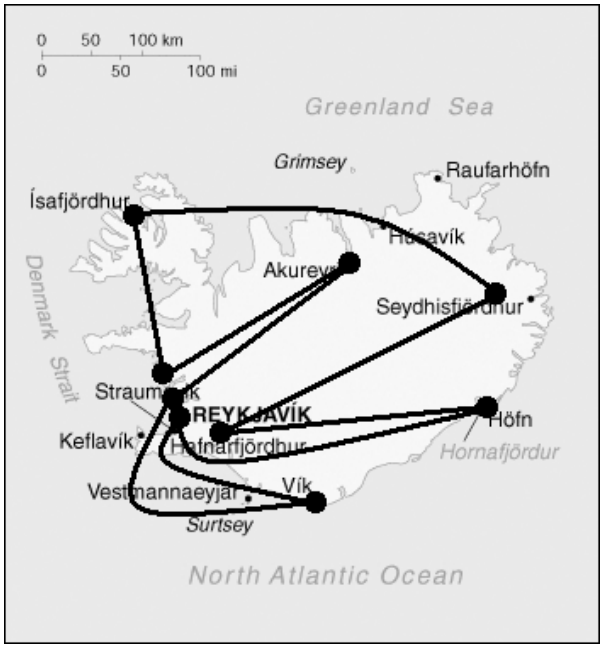

Figure 3. The "Iceland" city set with a random (naïve) initial condition (pathway) superimposed.

```
<Node Label="Hofn">
   <T S="Hofn" D="Akranes" Value="493" P="0.125"/>
   <T S="Hofn" D="Akureyri" Value="512" P="0.125"/>
   <T S="Hofn" D="Borgarnes" Value="519" P="0.125"/>
   <T S="Hofn" D="Egilstaddir" Value="247" P="0.125"/>
   <T S="Hofn" D="Isafjordur" Value="902" P="0.125"/>
   <T S="Hofn" D="Reykjavik" Value="459" P="0.125"/>
   <T S="Hofn" D="Selfoss" Value="402" P="0.125"/>
   <T S="Hofn" D="Vik" Value="273" P="0.125"/>
</Node>
```

Figure 4. "Iceland": List of Transitions (T) from the Source (S), Hofn, to all other legitimate Destinations (D) with transitional (node-node) Probabilities (P) determined from $P_i=1/(N-1)$, where N=#nodes (naïve assignment). "Value" is the XML attribute name for "d_{ij}" in km.

(7) Selection rate was based on relative cost for each gene. Suppose two genes G1 and G2 had cost 2500 and 3000, respectively. Then, the survival weight for G1 was proportional to 1/2500 and that for G3 proportional to 1/3000. Summing these for all genes and normalizing to 1.0 allowed selection of survival using iterations of Math.random(). For example, if G1 ended up with 3% of the total fitness, then, for example, in the 50 iterations of Math.random(), each value in the interval [0.00, 0.03) would select for one G1 offspring (expected value 1.5 offspring out of 50). (8) Initial conditions could be assigned by any of the means discussed above (Figure 3). (9) Finally, node-node transition costs offered a unique opportunity to improve the expected fitness and performance of the GA. The "naïve" assignment of these transitional probabilities is shown in Figure 4. The "proportional inverse distance" or "By Distance" weighting scheme of Equation 1 is shown in Figures 5-6 for the "Iceland" and "USA" problems.

```
<Node Label="Akranes">
   <T S="Akranes" D="Akureyri" Value="353" P="0.04"/>
   <T S="Akranes" D="Borgarnes" Value="38" P="0.37"/>
   <T S="Akranes" D="Egilstaddir" Value="617" P="0.02"/>
   <T S="Akranes" D="Hofn" Value="493" P="0.03"/>
   <T S="Akranes" D="Isafjordur" Value="422" P="0.03"/>
   <T S="Akranes" D="Reykjavik" Value="49" P="0.29"/>
   <T S="Akranes" D="Selfoss" Value="91" P="0.16"/>
   <T S="Akranes" D="Vik" Value="220" P="0.06"/>
</Node>
```

Figure 5. "Iceland": List of Transitions (T) from the Source (S) Hofn to all other legitimate Destinations (D) with transitional (node-node) Probabilities (P) determined from $P_i=1/(d_i*\Sigma_{j=1...N}(1/d_{ij}))$, where N=#nodes (Equation 1). "Value" is the XML attribute name for "d_{ij}" in km.

```
<Node Label="Boston">
   <T S="Boston" D="Atlanta" Value="1075" P="0.05"/>
   <T S="Boston" D="Charlotte" Value="841" P="0.06"/>
   <T S="Boston" D="Chicago" Value="1015" P="0.05"/>
   <T S="Boston" D="Dallas" Value="1770" P="0.03"/>
   <T S="Boston" D="Denver" Value="2003" P="0.02"/>
   <T S="Boston" D="Detroit" Value="834" P="0.07"/>
   <T S="Boston" D="Houston" Value="1858" P="0.03"/>
   <T S="Boston" D="Kansas City" Value="1437" P="0.04"/>
   <T S="Boston" D="Los Angeles" Value="3026" P="0.01"/>
   <T S="Boston" D="Minneapolis" Value="1425" P="0.04"/>
   <T S="Boston" D="New York" Value="211" P="0.23"/>
   <T S="Boston" D="Philadelphia" Value="314" P="0.15"/>
   <T S="Boston" D="Phoenix" Value="2690" P="0.02"/>
   <T S="Boston" D="St. Louis" Value="1187" P="0.04"/>
   <T S="Boston" D="San Antonio" Value="2044" P="0.02"/>
   <T S="Boston" D="San Diego" Value="3043" P="0.01"/>
   <T S="Boston" D="San Fran." Value="3140" P="0.01"/>
   <T S="Boston" D="Seattle" Value="3088" P="0.01"/>
   <T S="Boston" D="Washington" Value="442" P="0.11"/>
</Node>
```

Figure 6. "USA": List of Transitions (T) from the Source (S) Boston to all other legitimate Destinations (D) with transitional (node-node) Probabilities (P) determined from $P_i=1/(d_i*\Sigma_{j=1...N}(1/d_{ij}))$, where N=#nodes (Equation 1). "Value" is the XML attribute name for "d_{ij}" in miles.

3. Results and Discussion

Data were obtained for different initial conditions and/or node-node transition probabilities as follows: three sets of 1000 runs (computing time: 1-15 min/set) of the genertic GA software were performed, and the following data were obtained for these 1000 runs in each set: (1) minimum (C_{min}), maximum (C_{max}) and range (C_r) of "lowest cost", or "optimal" pathways obtained in each run; (2) mean (μ) of the optimal pathways obtained (or "error"), and the standard deviation (σ_μ) of the means for the three sets of 1000 runs; and (3) the number of iterations (N_i) to converge on the optimal value so obtained (along with the standard deviation of N_i, σ_N). Since for each of the two test problems, the true (global) optimum cost (C_{opt}) was known (2009 km for the "Iceland" problem, 9271 miles for the "USA" problem), C_{min}, C_{max}, C_r, μ and σ_μ were normalized by C_{opt} (and are thus presented as percentages of the optimal cost) and 100% subtracted from them (excepting C_r and σ_μ) so that they show incremental percentage over the optimum.

"Iceland" Test Problem

Using node-node probabilities to determine the initial conditions (initial set of genes) was assessed using several regimens. The first was a naïve assignment in which the next node was randomly selected from the remaining (legitimate) pool. The second used relative distance weighting for all nodes (normalized to 1.0 after each assignment for the remaining nodes) where step size $\Delta P=0.05$ and $P(Min)=0.02$, so $P(Max)=0.23$. The third used the results of clustering to pool together the nodes (Akranes, Borgarnes, Reykjavik and Selfoss) and thus exclude the other five nodes (within cluster weighting was increased to 3, 5, 15 and 30 times the extra-cluster weighting, with peak effectiveness at 15X, the reported value in Table 1). The fourth used the inverse cost (in this case cost=distance) weighting as in Equation 1. The results for these four tests are in Table 1.

	Naïve	Rel. Dist	Cluster	Distance
C_{min} (%)	0.00	0.00	0.00	0.00
C_{max} (%)	11.50	8.41	6.57	8.16
C_r (%)	11.50	8.41	6.57	8.16
μ (%)	**0.247**	**0.085**	**0.130**	**0.054**
σ_μ (%)	0.012	0.021	0.029	0.017
N_i	**8.049**	**5.930**	**6.434**	**4.592**
σ_N	0.027	0.076	0.165	0.116

Table 1. Results for the "Iceland" test problem with node-node transitions for determining the initial set of genes (combined with a roulette wheel) dictated by the "Naïve", "By Relative Distance", "By Cluster" and "By Distance" node-node transition probabilities. All groups are statistically significantly different in comparing (μ +/- σ_μ) and (N_i +/- σ_N) [using paired t-tests or ANOVA]. C_{min} = 0.00 for all sets of runs herein (justifying 1000 runs/set).

The results in Table 1 illustrate the utility of employing the inverse of the cost function to dictate successive nodes in the initial gene set. The naïve assignment (random gene sequencing) results in 0.247% expected increase in best solution cost over true optimum ("error"), with a mean of 8.05 iterations (representing 50*8.05 ~ 402 genes) to reach convergence. A substantial improvement is obtained by the simple clustering technique used: the mean optimal cost obtained is now only 0.130% above true optimum (a 47.4% relative decrease) with a concomitant decrease to 6.434 (a 20.1% relative decrease) iterations (~ 322 genes) to reach convergence. However, since distances within the cluster are treated as equal by this method, it is not surprising that further improvement is obtained by the "Relative Distance" method, in which longer distances from a node are incrementally weighted less (for probability). Here, the mean optimal cost obtained is now only 0.085% above true optimum (a 65.6% relative decrease from "Naïve") while reducing further (to 5.950, or ~298 genes, a 26.1% relative decrease from "Naïve") the iterations to reach convergence. Lastly, substantially improved results are obtained when initial gene node-node sequences are assigned "By Distance": mean optimal cost reduces to 0.054% above true optimum (78.1% relative decrease from "Naïve") and iterations to reach convergence reduce to 4.592 (~ 230 genes, a 42.9% relative decrease from "Naïve"). "Algorithmic Efficacy" (AE) can be defined as:

$$AE = k/(\mu * N_i) \qquad \textbf{Equation 4}$$

where k is a normalizing constant for the particular test problem (k=1 for the "Iceland" problem). AE for the "Naïve", "By Relative Distance", "By Cluster", and "By Distance" algorithms is 0.50, 1.98, 1.20 and 4.03, respectively. This value is a good metric for comparison, and indicates that the "By Distance" method is eight times as effective as the "Naïve" method.

Next, the initial conditions as described above ("Naïve", "Weighted Nearest Neighbor", "Lowest Remaining Distance", "Centroid + Clockwise / Counterclockwise Traversal" and "Clustering") were considered. Since the "Iceland" problem is relatively simple, the "Naïve" and "Clustering" initial conditions were the only ones that did not produce the global optimum immediately. When the "Clustering" initial condition was used, 10% (that is, 5) of the original gene set were defined as (Akranes, Borgarnes, Reykjavik, Selfoss) followed by (Akureyri, Egilstaddir, Hofn, Isafjordur, Vik); in other words, the clustered cities and non-clustered cities were simply sequenced alphabetically. When the runs were

performed, the values for (μ +/- σ_μ) and (N_i +/- σ_N) obtained were (0.219+/-0.032%) and (8.108+/-0.099), respectively, yielding an AE of only 0.56 (and indicating that N=9 may be too low for clusters).

The last set of experiments on the "Iceland" problem focused on the utility of the four node-node probability schemes described earlier (Table 1) when deployed for crossover splice location in addition to determining the initial gene set. These results are presented in Table 2.

	Naïve	Rel. Dist	Cluster	Distance
C_{min} (%)	0.00	0.00	0.00	0.00
C_{max} (%)	8.46	8.16	8.51	4.74
C_r (%)	8.46	8.16	8.51	4.74
μ (%)	**0.261**	**0.121**	**0.329**	**0.0066**
σ_μ (%)	0.058	0.010	0.040	0.0056
N_i	**8.167**	**5.371**	**6.196**	**3.056**
σ_N	0.052	0.068	0.130	0.138

Table 2. Results for the "Iceland" test problem with node-node transitions for determining both the initial gene set and the crossover locations (combined with a roulette wheel) dictated by the "Naïve", "By Relative Distance", "By Cluster" and "By Distance" node-node transition probabilities. "By Relative Distance" and "By Distance" groups are statistically significantly different from all other groups in comparing (μ +/- σ_μ) and (N_i +/- σ_N) [using paired t-tests or ANOVA].

The results in Table 2 indicate that the "Cluster" method applied to initial gene set and crossover provides no improvement in mean optimal cost compared to "Naïve", and indeed the AE (0.49 compared to 0.47 for "Naïve" in this Table, 0.50 in Table 1). The "Relative Distance" value for AE is 1.54, slightly worse than when it is used in determining the initial gene set only. However, the AE for the "By Distance" technique is 49.58, or 100 times the "Naïve" value (and 25 times the value obtained when "Distance" is used for the initial gene set only). Thus, the "By Distance" technique, in which node-node probabilities are inversely proportional to the node-node cost (distance), is far more effective than any of the other techniques investigated.

"USA" Test Problem

The same set of regimens was applied to the "USA" problem for determining the initial gene set (of 50). For the relative distance weighting, the step size ΔP=0.05 and P(Min)=0.00763, so P(Max)=0.09763. The clustering technique pooled the nodes (Los Angeles, San Diego), (Dallas, Houston, San Antonio), (Atlanta, Charlotte) and (Boston, New York, Philadelphia and Washington D.C.), thus excluding the other 9 cities (a relative cluster-to-noncluster weighting of 25X is reported in Table 3).

Because the "USA" problem is innately more challenging than the "Iceland" problem, the values in Table 3 are higher for C_{max}, C_r, μ and N_i. As a consequence, for AE (Equation 4) in the "USA" problem, using k=1000 is useful. For the "Naïve" initial gene set assignment, AE = 1.58. "By Relative Distance" AE = 1.66, "By Cluster" AE = 1.60, and "By Distance" AE = 1.62. These values are similar, and indicate that for more complex problems, initial gene set assignment may not have a particularly strong effect on algorithmic efficacy.

The effect of initial conditions ("Naïve", "Weighted Nearest Neighbor", "Lowest Remaining Distance", "Centroid + Clockwise / Counterclockwise Traversal" and "Clustering") were considered. When the "Lowest Remaining Distance" initial condition was used, the values for (μ +/- σ_μ) and (N_i +/- σ_N) obtained were (6.223+/-0.095%) and (99.61+/-3.76), respectively, yielding an AE of 1.61. However, the first noticeable improvement in algorithmic efficacy for the "USA" problem occurred when the "Centroid + Clockwise / Counterclockwise Traversal" initial condition was used for 10% of the initial gene set. When this pathway (which has a supra-optimal traversal distance of 11100 miles, or +19.7%) was used together with "Naïve" node-node probabilities, the values for (μ +/- σ_μ) and (N_i +/- σ_N) obtained were (3.107+/-0.089%) and (52.45+/-2.73), respectively, yielding an AE of 6.14, or a four-fold improvement over the

previous methods. In the next set of runs, this pathway was used together with "By Distance" node-node probabilities, and the values for (μ +/- σ_μ) and (N_i +/- σ_N) obtained were (2.270+/-0.016%) and (81.69+/-0.97), respectively, yielding an AE of 5.39, with a significantly lower μ than for any previous methods.

	Naïve	Rel. Dist	Cluster	Distance
C_{min} (%)	0.00	0.00	0.00	0.00
C_{max} (%)	33.00	29.08	28.63	28.51
C_r (%)	33.00	29.08	28.63	28.51
μ (%)	**9.866**	**8.094**	**7.108**	**6.234**
σ_μ (%)	0.460	0.292	0.151	0.031
N_i	**63.99**	**74.57**	**87.69**	**99.04**
σ_N	0.87	1.17	1.29	2.56

Table 3. Results for the "USA" test problem with node-node transitions for determining the initial set of genes (combined with a roulette wheel) dictated by the "Naïve", "By Relative Distance", "By Cluster" and "By Distance" node-node transition probabilities. All groups are statistically significantly different in comparing (μ +/- σ_μ) and (N_i +/- σ_N) [using paired t-tests or ANOVA].

The last set of experiments on the "USA" problem also focused on the utility of the node-node probability schemes when deployed for crossover splice location in addition to determining the initial gene set. The same set as described for Table 2 was used with the more complicated "USA" problem, and the results are presented in Table 4. Note that the first columns in Tables 3 and 4 (as for Tables 1 and 2) should be statistically similar/equivalent (and are), since they represent the same "Naïve" protocols. Unlike the results in Table 3, the results in Table 4 show compelling changes in μ without offsetting increases in N_i. Thus, AE (with k=1000) is 1.64, 3.99, 4.63 and 11.87, respectively, for the four columns, and the "error" μ improves significantly by 2.46, 2.81 and 6.48 times for the last three columns, respectively, when compared to the "Naïve" results.

	Naïve	Rel. Dist	Cluster	Distance
C_{min} (%)	0.00	0.00	0.00	0.00
C_{max} (%)	52.00	23.55	17.58	11.50
C_r (%)	52.00	23.55	17.58	11.50
μ (%)	**9.557**	**3.886**	**3.407**	**1.475**
σ_μ (%)	0.113	0.077	0.074	0.042
N_i	**63.97**	**64.52**	**63.46**	**57.11**
σ_N	0.86	1.07	1.67	0.07

Table 4. Results for the "USA" test problem with node-node transitions for determining both the initial gene set and the crossover locations (combined with a roulette wheel) dictated by the "Naïve", "By Relative Distance", "By Cluster" and "By Distance" node-node transition probabilities. All groups are statistically significantly different from all other groups in comparing (μ +/- σ_μ); and the "By Distance" group is statistically significantly lower than the other groups for (N_i +/- σ_N) [using paired t-tests or ANOVA].

Overall Considerations

Clearly, the hybrid genetic program used herein, combining GA search and convergence efficiency with pragmatic costs for node-node transitions, is an effective means to improve both algorithmic "error" (in terms of the expected value of the converged solution compared to the optimal solution) and the iterations required for convergence. For the two navigational test problems presented herein, the use of "By Distance" weighting of node-node transitions for both initial gene sequencing and for dictating crossover splicing locations results in compelling improvements in both "error" and iterations. In the "Iceland" problem, the method of Equation 1 reduced error by a factor of 39.5 while simultaneously decreasing the number of iterations by a factor of 2.7. In the more complicated "USA" example, the method of Equation 1 reduced the error by a factor of 6.5 while simultaneously decreasing the number of iterations by a factor of

1.12. The relative value of the other (simpler) distance-based methods depends on the nature of the test problem: the "Iceland" problem, for example, is optimized by a (counter)clockwise traversal around the centroid while the "USA" problem is not. The "USA" problem, on the other hand, responds better (in relative terms) to the clustering approaches. Regardless, the data support the hypothesis that "the more the initial gene sequences and crossover splices are based on the GA's cost function (distance in this case), the more increase in efficacy is observed." This intuitive hypothesis has been herein proven highly effective in direct application to GA's, since there are effective means (via initial gene sequence and crossover) to apply these costs directly to the iterative, evolving solution domain of GA's.

The use of a traditional GA with a weighted scheme for crossover and/or initial conditions based on proximity of nodes (in terms of node-node costs) is considered to be a hybrid genetic program whereby real-world (or "fact-based") cost considerations are used to guide the GA's solutions while still benefiting from GA's searching efficiency. The results are applicable to many broad areas of navigation: visiting a set of destinations, finding best paths when a particular destination must be replaced or a particular node-node transition is missing (e.g. detours, closed roads, traffic jams, and other "real-world" applications), and even extra-navigational process optimizations. Detours can be represented in at least two ways: (1) implicitly as prohibitively high node-node costs (or, equivalently, zero-valued node-node probabilities), and (2) explicitly as "missing" transitions. For the former, we can represent for example the Hofn-to-Egilstaddir transition in Figure 4 as:

<T S="Hofn" D="Egilstaddir" Value="247" P="0.0"/>

while for the latter is can be simply omitted from the description of the <Node Label="Hofn"> element.

In addition, the procedure outlined herein can be applied to the immense set of problems whereby point-to-point (node-node, locus-locus, process-process, etc.) costs can be estimated either *a priori* or real-time. Navigation is an obvious, valuable domain for application, and the approaches used here are clearly designed with navigation in mind. Further "hybridization" in the navigational realm can include adding prescribed pathways (i.e. "you must travel to Boston after you travel to Chicago") to the solution, not requiring a full circuit, or allowing nodes to be omitted. Further areas of application include many processes (each of which can have a transition cost), including client-server transaction design, distributed computing, and navigational-related delivery and supply chain processes. Extended to workflows, hybrid GA's may be valuable in determining the sequence of events. The work presented herein is the starting point for a rich realm of such applications and experiments.

Gene linkage and related co-inheritance techniques [4-7] can be viewed as similar to the clustering techniques described herein where $p(C_i,N_j) = 0.0$ for all nodes i in the cluster C and all nodes j in the nonclustered space S = N-C (N is the set of all nodes). However, the work presented here automates the linkage of loci/genes in a non-binary and adaptive fashion, and its performance is significantly enhanced compared to clustering schemes. Loci pairs with lower costs of transition are innately linked to a greater degree than loci pairs with higher transition costs. In other words, the "gene linking" comes for free.

4. Future Work

Hybrid-GA optimization is of interest, and the work here only begins to address this issue. Future work includes the need for larger, more diverse problem sets. Traditional GA test problems such as the Royal Road and One-Max [4] are not appropriate for the hybrid GA-based programming (HGAP) technique herein introduced.

Further, more in-depth, qualitative exploration of the advantages and limitations of the HGAP technique are necessary. Separate testing of different crossover strategies and rates, mutation strategies and rates, and more advanced "jiggling" strategies are required. All of these can benefit from ongoing work in traditional GA's, with or without gene linking. Additionally, more complicated cost functions than simple distance should be explored using HGAP techniques. Equation 3 provides the method to optimize node-node transition probabilities when the cost function is more complicated.

This paper introduces an important metric for comparisons between different HGAP methods: "Algorithmic Efficacy" (Equation 4). This provides a good figure of merit, incorporating error and iteration. However, other valuable comparative metrics are likely to be useful in navigational and other domains in which HGAP's can be applied.

Conditional (Bayesian) methods of assigning initial state and crossover splicing locations were introduced but not evaluated here. It is likely that in many navigational problems error and convergence rate improvements can be garnered by employing "By Direction" and other Bayesian techniques to the HGAP initial state, crossover, and mutation parameters. Rule-based direction of initial state and selection (e.g. spectacularly unlikely genes are discarded) should also be evaluated.

Finally, since the HGAP technique is innately related to cost optimization, a more mature cost representation than simple node-node transition weighting is critical. When non-exhaustive (non-"traveling salesman") pathways are legitimate, the nodes themselves must be weighted, and cost optimization will involve several (often competing) factors: node value, node-node transition cost, replacement node value, multi-node value schema (i.e. where visiting a particular set of nodes sequentially or during the entire pathway increases their summed value), and possible additional value for re-visiting a node, among others. Such multi-layered problems provide an exciting opportunity for the hybrid of GA and cost-based techniques.

5. References

[1] R.P.N. Rao, O. Fuentes, "Hierarchical learning of navigational behaviors in an autonomous robot using a predictive sparse distributed memory," *Autonomous Robots* 5(3-4):297-316, 1998 and *Machine Learning* 31(1-3):87-113, 1998.

[2] D.E. Goldberg, *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley Publishing Company, Inc., Reading, Mass, 412 pp., 1989 (ISBN 0-201-15767-5).

[3] L.A. Laane, "Development of navigational controllers for vehicles in highway traffic situations via genetic programming," In *Genetic Algorithms and Genetic Programming at Stanford 1995*, J.R. Koza (ed.), Stanford Bookstore, 171-180, 1995 (ISBN 0-18-195720-5).

[4] J. Smith, "On appropriate adaptation levels for the learning of gene linkage", *Genetic Programming and Evolvable Machines* 3:129-155, 2002.

[5] A. Salman, K. Mehrota and C. Mohan, "Linkage crossover for genetic algorithms," In *Proc. Genetic and Evolutionary Computation Conf.*, W. Banzhaf et al. (eds.), 564-571, 1999.

[6] J. Schaffer and A. Morishima, "An adaptive crossover distribution mechanism for genetic algorithms," In *Proc. Second Internat. Conf. Genetic Algorithms*, J.J. Grefenstette (ed.), 36-40, 1987.

[7] J.D. Schaffer and L.J. Eshelman, "On crossover as an evolutionary viable strategy," In *Proc. Fourth Internat. Conf. Genetic Algorithms*, R. Belew and L. Booker (eds.), 61-68, 1991.

[8] D.E. Goldberg, B. Korb and K. Deb, "Messy genetic algorithms: Motivation, analysis, and first results," *Complex Systems* 3(5):493-530, 1989.

[9] H. Beyer, "Toward a theory of evolution strategies: On the benefit of sex-the (μ/μ-λ)-strategy", *Evolutionary Computation* 3:81-111, 1995.

[10] Ísland Fer∂akort, Mál og menning, su∂urlandsbraut 12, IS-108 Reykjavik, http://www.edda.is, 2002.

[11] http://www.freegk.com/worldatlas/iceland.php.

[12] *Road Atlas (United States, Canada, Mexico)*, Rand McNally, page A4, 1997 (ISBN 0-528-81555-5).